

### Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

#### Listing of Claims:

---

Claim 1 (original): A method of providing checksum values for data segments retrieved from a data storage device for transfer into a buffer memory, comprising the steps of:

- (a) maintaining a checksum list comprising a plurality of entries  
5 corresponding to the data segments stored in the buffer memory, each entry for storing a checksum value for a corresponding data segment stored in the buffer memory; and
- (b) for each data segment retrieved from the storage device:
- (1) calculating a checksum value for that data segment using a  
checksum circuit for performing a checksum function;
- (2) storing the checksum value in an entry in the checksum list;  
10 and
- (3) storing the data segment in the buffer memory;

wherein, for transferring packets of data out of the buffer memory, a checksum value can be calculated for data in each packet based on one or more checksum values  
15 stored in the checksum list.

Claim 2 (original): The method of claim 1, wherein step (b)(1) further includes the steps of calculating the checksum for said data segment using the checksum circuit as the data segment is transferred into the buffer memory.

Claim 3 (original): The method of claim 1 further comprising the steps of:

- (c) building packets of data for transfer out of the buffer memory, and
- (d) providing a checksum value for data in each packet based on one or more checksum values stored in the checksum list.

Claim 4 (original): The method of claim 3, wherein:

each packet in a set of said packets includes one or more complete data segments; and

the step (d) of providing a checksum value for each of the packets in said set of packets further includes the steps of:

(1) retrieving the checksum value for each data segment in that packet from a corresponding entry in the checksum list; and

(2) calculating a checksum value for that packet as a function of the retrieved checksum values.

Claim 5 (original): The method of claim 3, wherein:

each packet in a set of said packets includes: (i) one or more complete data segments, and (ii) a fragment of each of one or more data segments; and

the step (d) for providing a checksum value for each of the packets in said  
5 set of packets further includes the steps of:

(1) retrieving the checksum value for each of said one or more  
complete data segments in that packet from a corresponding entry in the checksum list;

(2) for each data segment fragment in that packet, calculating a  
checksum value for data in that data segment fragment; and

10 (3) determining a checksum value for all of the data in that  
packet as a function of the retrieved and calculated checksum values.

B1  
Claim 6 (original): The method of claim 5 further comprising the steps of:

for at least one packet in step (d), caching one or more of said calculated  
checksum values for the data segment fragments in that packet, wherein the cached  
checksum values can be used for determining a checksum value for a subsequent data  
5 packet.

Claim 7 (original): The method of claim 6 wherein:

for at least one data segment fragment in each of one or more of said set of  
packets in step (d), the step of calculating a checksum value for that data segment  
fragment in step (d)(2) further includes the steps of calculating the checksum value for  
5 that data segment fragment as a function of one or more previously cached checksum  
values.

Claim 8 (original): The method of claim 3, wherein:

each packet in a set of said packets includes: (i) at least one complete data segment, and (ii) a fragment of each of one or more data segments; and

the step (d) for providing a checksum value for data in each of the packets  
5 in said set of packets further includes the steps of:

(1) retrieving the checksum value for said at least one complete data segment in that packet from a corresponding entry in the checksum list;

(2) for each of said one or more data segment fragments in the packet, retrieving the checksum value from the entry in the checksum list corresponding  
10 to the data segment of which that data segment fragment is a part, and calculating a checksum value for data in a complementary fragment of that data segment; and

(3) determining a checksum value for all of the data in that packet as a function of the retrieved and calculated checksum values.

Claim 9 (original): The method of claim 8, wherein step (d)(3) includes the steps  
of:

calculating an intermediate checksum value as a function of the retrieved checksum values; and

5 adjusting the intermediate check sum value using the calculated checksum values to provide the checksum value for all the data in that packet.

Claim 10 (original): The method of claim 8 further comprising the steps of:

for at least one packet in step (d), caching one or more of said calculated checksum values in step (d)(2), wherein the cached checksum values can be used for determining a checksum value for a subsequent data packet.

Claim 11 (original): The method of claim 10 wherein:

for at least one data segment fragment in each of one or more of said set of packets in step (d), step (d)(2) further includes the steps of calculating the checksum value of said complementary fragment as a function of one or more previously cached checksum values.

5

Claim 12 (currently amended): The method of claim 3, wherein:

each packet in a set of said packets includes: (i) at least one complete data ~~segments~~ segment, and (ii) a fragment of each of one or more data segments; and

the step (d) for providing a checksum value for each of the packets in said set of packets further includes the steps of:

5

(1) retrieving the checksum value for said at least one complete data segment in that packet from a corresponding entry in the checksum list;

(2) for each of said one or more data segment fragments in the packet, if that data segment fragment is smaller in size than a complementary fragment of the data segment of which that data segment fragment is a part, then calculating a checksum value for data in that segment fragment, otherwise, retrieving the checksum value from the entry in the checksum list corresponding to said data segment of which

10

that data segment fragment is a part, and calculating a checksum value for data in said complementary fragment of that data segment; and

15 (3) providing a checksum value for the data in that packet as a function of said retrieved checksum values and said calculated checksum values.

Claim 13 (original): The method of claim 12 further comprising the steps of:

for at least one packet in step (d), caching one or more of said calculated checksum values in step (d)(2), wherein the cached checksum values can be used for determining a checksum value for a subsequent data packet.

B1

Claim 14 (original): The method of claim 13, wherein for at least one data segment fragment in each of one or more of said set packets in step (d), step (d)(2) further includes the steps of:

5 if that data segment fragment is smaller in size than a complementary fragment of the data segment of which that data segment fragment is a part, then calculating a checksum value for data in that data segment fragment as a function of one or more previously cached checksum values;

10 otherwise, retrieving the checksum value from the entry in the checksum list corresponding to said data segment of which that data segment fragment is a part, and calculating a checksum value for data in said complementary fragment of that data segment as a function of one or more previously cached checksum values

Claim 15 (original): The method of claim 3, wherein:

each packet in a set of said packets includes a fragment of each of one or more data segments;

the steps of providing a checksum value for each of the packets in said set  
5 of packets further includes the steps of:

for each fragment of a data segment in that packet, calculating a checksum value for data in that data segment fragment; and

determining a checksum value for all of the data in that packet as a function of the calculated checksum values.

Claim 16 (original): The method of claim 15, wherein for at least one data  
segment fragment, the step of calculating a checksum includes the steps of:

retrieving the checksum value from the entry in the checksum list  
corresponding to the data segment of which that data segment fragment is a part;

5 calculating a checksum value for data in a complementary fragment of that data segment; and

determining the checksum value for that data segment fragment as a function of: (i) said retrieved checksum value, and (ii) said calculated checksum value for data in the complementary fragment.

Claim 17 (original): A storage device comprising:

- (a) storage media for storing data segments;
- (b) buffer memory; and

(c) a checksum circuit for providing checksum values for data  
5 segments retrieved from the storage media for transfer into the buffer memory, and for  
storing the checksum values in a checksum list including a plurality of entries  
corresponding to the data segments stored in the buffer memory, each entry for storing a  
checksum value for a corresponding data segment in the buffer memory;

wherein, for transferring packets of data out of the buffer memory, a checksum  
10 value can be calculated for data in each packet based on one or more checksum values  
stored in the checksum list.

Claim 18 (original): The storage device of claim 17, wherein the checksum  
circuit further comprises:

- B1
- (i) a logic circuit for calculating a checksum value for a data segment;
  - (ii) means for locating an entry in the checksum table corresponding to  
5 that data segment; and
  - (iii) means for storing the checksum value in the located entry in the  
checksum list.

Claim 19 (currently amended): The storage device of claim 17, further  
comprising a microcontroller ~~configured by program instruction for building that builds~~  
packets of data for transfer out of the buffer memory, and ~~for providing~~ provides a  
checksum value for data in each packet based on one or more checksum values stored in  
5 the checksum list.



Claim 20 (currently amended): The storage device of claim 19, wherein:  
~~the microcontroller is further configured by program instructions for building~~  
~~packets of data for transfer out of the buffer memory~~, each packet in a set of said packets  
~~including~~ includes one or more complete data segments, and

5        ~~for providing~~ the microcontroller further provides a checksum value for each of  
the packets in said set of packets by retrieving the checksum value for each data segment  
in that packet from a corresponding entry in the checksum list and calculating a  
checksum value for that packet as a function of the retrieved checksum values.

Claim 21 (currently amended): The storage device of claim 19, wherein:

~~the microcontroller is further configured by program instructions for:~~

(B)  
~~building packets of data for transfer out of the buffer memory~~, each packet  
in a set of said packets ~~including~~ includes one or more complete data segments, and a  
5        fragment of each of one or more data segments; and

the microcontroller further provides ~~providing~~ a checksum value for each  
of the packets in said set of packets by:

(i)        retrieving the checksum value for each of said one or more  
complete data segments in that packet from a corresponding entry in the checksum list;

10        (ii)       for each data segment fragment in that packet, calculating a  
checksum value for data in that data segment fragment; and

(iii)       determining a checksum value for all of the data in that  
packet as a function of the retrieved and calculated checksum values.

Claim 22 (currently amended): The storage device of claim 19, wherein:

~~the microcontroller is further configured by program instructions for:~~

~~building packets of data for transfer out of the buffer memory, each packet~~

in a set of said packets ~~including~~ includes at least one complete data segment, and a

5 fragment of each of one or more data segments; and

the microcontroller further provides ~~providing~~ a checksum value for data  
in each of the packets in said set of packets by:

(i) retrieving the checksum value for said at least one complete  
data segment in that packet from a corresponding entry in the checksum list;

10 (ii) for each of said one or more data segment fragments in the  
packet, retrieving the checksum value from the entry in the checksum list corresponding  
to the data segment of which that data segment fragment is a part, and calculating a  
checksum value for data in a complementary fragment of that data segment; and

(iii) determining a checksum value for all of the data in that  
15 packet as a function of the retrieved and calculated checksum values.

Claim 23 (currently amended): The storage device of claim 19, wherein:

~~the microcontroller is further configured by program instructions for:~~

~~building packets of data for transfer out of the buffer memory, each packet~~

in a set of said packets ~~including~~ includes a fragment of each of one or more data

5 segments; and

the microcontroller further provides ~~providing~~ a checksum value for each  
of the packets in said set of packets ~~further includes the steps of~~ by:

for each fragment of a data segment in that packet, calculating a checksum value for data in that data segment fragment; and

10 determining a checksum value for all of the data in that packet as a function of the calculated checksum values.

Claim 24 (currently amended): The storage device of claim 23, wherein ~~the microcontroller is further configured by program instructions such that~~ for at least one data segment fragment, ~~the step of calculating a checksum value includes the steps of~~ microcontroller further:

5 ~~retrieving~~ retrieves the checksum value from the entry in the checksum list corresponding to the data segment of which that data segment fragment is a part;

calculating calculates a checksum value for data in a complementary fragment of that data segment; and

determining determines the checksum value for that data segment  
10 fragment as a function of: (i) said retrieved checksum value, and (ii) said calculated checksum value for data in the complementary fragment.

Claim 25 (original): A checksum system for a data processing system including at least one data storage device and buffer memory, the checksum system being for providing checksum values for data segments retrieved from the data storage device for transfer into the buffer memory, the checksum system comprising:

5 (a) a checksum list including a plurality of entries corresponding to the data segments stored in the buffer memory, each entry for storing a checksum value for a corresponding data segment stored in the buffer memory;

(b) a logic circuit for calculating a checksum value for each data segment;

10 (c) means for locating an entry in the checksum table corresponding to that data segment; and

(d) means for storing the checksum values in the located entry in the checksum list;

wherein, for transferring packets of data out of the buffer memory, a checksum value can be calculated for data in each packet based on one or more checksum values stored in the checksum list.

(B)  
Claim 26 (currently amended): The checksum system of claim 25 further comprising a processor configured ~~by program instructions~~ for building packets of data for transfer out of the buffer memory, and for providing a checksum value for data in each packet based on one or more checksum values stored in the checksum list.

Claim 27 (currently amended): The checksum system of claim 26, wherein:  
~~the processor is further configured by program instructions for building packets of data for transfer out of the buffer memory,~~ each packet in a set of said packets ~~including~~ includes one or more complete data segments, and

5            the processor is further configured to provide for providing a checksum value for each of the packets in said set for packets by retrieving the checksum value for each data segment in that packet from a corresponding entry in the checksum list and calculating a checksum value for that packet as a function of the retrieved checksum values.

Claim 28 (currently amended): The checksum system of claim 26, wherein:

~~the processor is further configured by program instructions for:~~

~~building packets of data for transfer out of the buffer memory,~~ each packet in a set of said packets ~~including~~ includes one or more complete data segments, and a  
5       fragment of each of one or more data segments; and

the processor is further configured to provide ~~providing~~ a checksum value for each of the packets in said set of packets by:

             (i)       retrieving the checksum value for each of said one or more complete data segments in that packet from a corresponding entry in the checksum list;

10               (ii)       for each data segment fragment in that packet, calculating a checksum value for data in that data segment fragment; and

             (iii)       determining a checksum value for all of the data in that packet as a function of the retrieved and calculated checksum values.

Claim 29 (currently amended): The checksum system of claim 26, wherein:

~~the processor is further configured by program instructions for:~~

~~building packets of data for transfer out of the buffer memory~~, each packet  
in a set of said packets ~~including~~ includes at least one complete data segment, and a  
5 fragment of each of one or more data segments; and

the processor is further configured to provide ~~providing~~ a checksum value  
for data in each of the packets in said set of packets by:

(i) retrieving the checksum value for said at least one complete  
data segment in that packet from a corresponding entry in the checksum list;

10 (ii) for each of said one or more data segment fragments in the  
packet, retrieving the checksum value from the entry in the checksum list corresponding  
to the data segment of which that data segment fragment is a part, and calculating a  
checksum value for data in a complementary fragment of that data segment; and

(iii) determining a checksum value for all the data in that packet  
15 as a function of the retrieved and calculated checksum values.